Hindawi
Journal of Robotics
Volume 2023, Article ID 6657120, 13 pages
https://doi.org/10.1155/2023/6657120

Research Article

Research on Target Tracking Algorithm of Micro-UAV Based on Monocular Vision

Yingbin Feng , Dian Wang , and Kun Yang

School of Automation and Electrical, Shenyang Ligong University, Shenyang, Liaoning 110159, China

Correspondence should be addressed to Yingbin Feng; 781787793@qq.com

Received 14 February 2023; Revised 29 May 2023; Accepted 31 May 2023; Published 4 July 2023

Academic Editor: Bingxiao Ding

Aiming at the problem of limited payload and endurance of micro-UAV, the target tracking algorithm based on monocular vision is proposed. Since monocular vision cannot directly measure distance between the UAV and the target, triangulation and triangle similarity are used to calculate the distance information. Then, a target tracking method based on Kalman filter and KCF is designed. The tracking result of KCF is modified by Kalman filter to solve the problem of target occlusion. Finally, the position of the target in the world coordinate system is calculated through the coordinate transformation matrix, which is used to control the UAV for tracking the moving target. In order to verify the feasibility of the algorithm, target size estimation and target tracking algorithms are carried out. The experimental results show that the proposed algorithm can track the moving target effectively under the condition of short-term occlusion.

1. Introduction

In recent years, with the rapid development of vision technology, communication technology, and flight control technology, unmanned aerial vehicles (UAVs) have been widely used in real-time monitoring, investigation, traffic control, and civil photography [1–5]. According to the UAV dimension, the UAV can be classified into micro-UAV, small-UAV, and large-UAV. Due to its small dimension, light weight, good mobility, and strong concealment, the microunmanned aerial vehicle (MAV) has unique advantages in target tracking [6]. However, the MAV is limited by payload and endurance, making it impossible to carry a large computer and huge detection sensors. One of the hot research problems is how to study accurate and robust target tracking algorithms for MAV platform.

Compared with other detection sensors, cameras with optical sensors as the core component receive more institution feedback on environmental information and key points. Moreover, the camera has the characteristics of low cost and light weight, so it has great potential in the field of target tracking. Cameras with optical sensors as the core component can be classified into monocular cameras, binocular cameras, and depth cameras based on the sensors they carry. All of these types of cameras have been used in target tracking. Aiming at the problem of inaccurate acquisition of depth images caused by UAV jitter, Tayyab Naseer's team of Technical University of Munich presented to simultaneously carry depth camera, monocular camera, and other sensors in the UAV system. And the team used a monocular camera and label positioning methods to assist the depth camera to obtain accurate depth image information for human motion tracking [7]. However, the system is currently only suitable for indoor environments and small-scale movements. Liu et al. presented to use a UAV equipped with a three-axis pan-tilt for tracking the target, which could filter the noise caused by UAV jitter and expand the field of view [8]. However, due to the large size of the three-axis pan-tilt, it cannot be carried on a MAV.

Target tracking algorithms can be divided into generative methods and discriminant methods. The generative methods only focus on the target feature, ignore the background information, and match the detected images by establishing a target model. Discriminant methods find the optimal region in the next frame of image by training a classifier to achieve the purpose of target tracking. The generative methods assume that the target features remain constant for a period, so these methods cannot track the target motion in complex situations. Discriminant methods based on correlation filter and deep learning can adapt to complex application scenarios.

In [9], the researcher used the correlation filtering algorithm to track the target and presented a minimum output sum of squared error (MOSSE) algorithm. The tracking speed of this algorithm can reach more than 600 frames per second, and it has the function of resisting illumination and the shape change of the target, which improves the tracking robustness. Henriques et al. presented the Kernelized Correlation Filter (KCF), which replaces the gray features of the original filtering method with histograms of oriented gradients (HOG) features [10]. Furthermore, the nonlinear classification problem is mapped to a high-dimensional space to make it linearly separable, and the computational complexity is reduced by applying kernel functions and the diagonalizable properties of circulant matrix. In order to solve the edge effect, Danelljan et al. presented a spatially regularized discriminative correlation filters (SRDCF) algorithm [11]. In [12], the researchers used real shifts to generate negative samples, used real samples to train filters, and expanded the search area to improve the tracking effect. However, the algorithm is easy to lose the target when the appearance of the target changes greatly. In order to further improve the performance of the correlation filter tracking algorithm, many algorithms extract deep features to represent the target [13, 14]. Although the tracking effect is improved, the tracking speed of the correlation filter algorithm based on deep features is slow and not suitable for the computing resources of the UAV platform. Aiming at the problem of background noise generated by UAV in flight, Huang et al. presented an aberrance repressed correlation filter (ARCF) algorithm, and the experiment results show that ARCF performs well on most UAV data sets [15]. However, it is difficult to effectively deal with tracking failure caused by target occlusion and size change.

With the rise of deep neural network technology, it has received extensive attention in the field of target tracking. Convolution neural network has strong target expression ability because of the deep features obtained by learning, which gradually replaces traditional manual features. It has been introduced into the target tracking task and has made great progress [16–18]. Siamese instance search tracker (SINT) creatively uses Siamese neural network to measure the similarity between template images and search images, which provides a new idea for target tracking [19]. To solve the problem of poor real-time performance of deep learning in target tracking, Bertinetto et al. proposed the fully-convolutional Siamese network (SiamFC) algorithm [20]. Due to the complex network structure of the deep learning tracking algorithm, it cannot achieve both speed and accuracy to a certain extent. In [21], the researchers presented a Siamese region proposal network (Siam-RPN) tracking algorithm. Due to the limited data set, the training quality of the Siam-RPN network is not high. Aiming at the tracking accuracy problem of Siam-RPN, Yu et al. presented a distractor-aware Siamese region proposal networks (DaSiamRPN) tracking algorithm based on Siam-RPN, which improved the anti-interference and discrimination ability of tracking and achieved a tracking speed of 160 frames per second [22]. Although deep learning tracking algorithms have made great progress, the lack of training samples makes it difficult to train high-quality neural networks for different tracking scenarios. In addition, deep neural networks have very high requirements for computer hardware resources, which also affect the application of the MAV platform.

In summary, the MAV target tracking mainly faces the following challenges:

(1) Limited by the structural characteristics of the MAV, ensuring target tracking accuracy and reducing the complexity of the algorithm are key problems that need to be resolved

(2) During the flight of UAV, the airframe jitter may cause camera shake, target blur, and other problems.

In addition, there may be short-term obstacles between the UAV and the target, which will lead to target drift and loss in tracking. Therefore, it is difficult to achieve stable and robust tracking of the UAV. This paper proposes a target tracking algorithm of MAV based on monocular vision to solve the abovementioned problems. Firstly, aiming at the problem that monocular camera cannot measure the depth information between the UAV and the tracking target, the initialization method of triangulation is proposed to measure the target size. Then, the triangle similarity method is applied to estimate the depth between the target and the camera to solve the two-dimensional limitation of the monocular camera. Secondly, aiming at the deficiencies of the KCF filter algorithm, a target tracking algorithm based on Kalman filter and KCF fusion is proposed. The tracking results of KCF are corrected by Kalman filter to improve the tracking accuracy and robustness. Finally, the position of the target in the world coordinate system is calculated by the coordinate transformation matrix, which is used as the expected input of the position to control the UAV to track the moving target.

2. System Architecture

In order to perform the tracking task, the UAV carries the monocular camera for image acquisition. As the optical flow sensor can measure the horizontal velocity of the UAV, the UAV usually uses it to achieve fixed-point flight indoors, and it also can be used in conjunction with GPS in outdoor environments. In addition, the Nvidia Jetson Nano is applied as an onboard computer; its Quad-core ARM A57 CPU and 4 GB RAM can fully meet the experimental requirements. The compact size of 100 mm × 80 mm × 29 mm can perfectly adapt to the size of the UAV. For flight control system, the UAV utilizes Holybro Pixhawk 4 as the UAV attitude control unit. Its PX4 firmware can run Offboard mode and execute upper control instructions. The UAV target tracking system is shown in Figure 1.

Concerning software, the robot operating system (ROS) is installed on the airborne computer to establish communication connections between multinodes, multitasks, and multiprocesses. The software mainly includes the following modules: (1) target tracking module fused with KCF and Kalman filter, (2) target position calculation, (3) position control, (4) the data

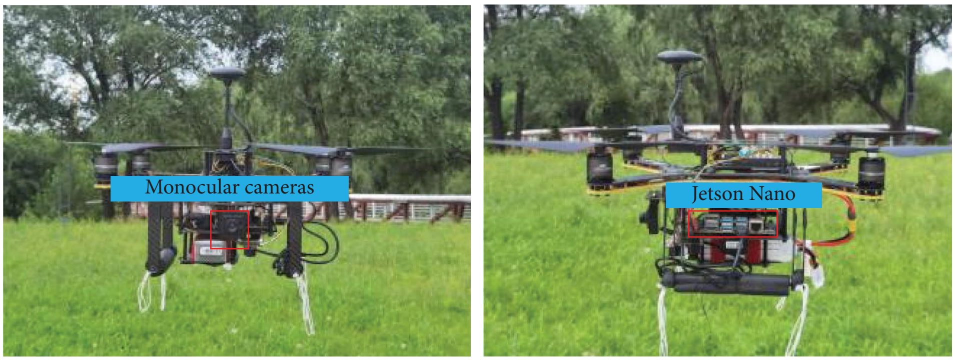


Figure 1: UAV target tracking system.

collection module for sensor, and (5) MAVROS software package. The UAV acquires images of the tracking target through the monocular camera; the fusion KCF and Kalman filter are used to track the dynamic target. The three-dimensional motion information of the target is calculated by position solution and sent to the flight control as the expected input of the position controller to perform the target tracking task. In the meantime, the QGroundControl (QGC) and the remote desktop can monitor the flight attitude and mission command of the UAV in real time. The software architecture is shown in Figure 2.

3. State Estimation of the Target

The prerequisite for performing target tracking is to estimate the position motion information of the target. The target tracker based on discriminant is used to generate the 2D motion information of the target in the image, and then the Kalman filter is established to fuse the abovementioned 2D motion information to obtain the final target tracking result.

3.1. The KCF Target Tracking Algorithm. The KCF (kernelized correlation filters) algorithm is a discriminative target tracking algorithm based on online learning. The initial frame is used to generate training sample sequences through circulant matrix shift. The target is detected by the ridge regression training classifier, and the area with the largest response is the target area. Although the KCF algorithm needs to generate multiple virtual samples through circulant matrix in the process of target tracking, there are plenty of matrix inversion calculations in the process of training the classifier. The algorithm makes use of the property that the circulant matrix can be diagonalized and applies the discrete Fourier matrix to diagonalize the sample set. Due to the diagonal matrix operation only needing to calculate the nonzero elements on the diagonal line, it can greatly reduce the occupation of CPU and memory resources. In addition, the KCF algorithm introduces the Gaussian kernel function to map the nonlinear problem to the high-dimensional space and converts it to the linear problem, which greatly improves the calculation speed and meets the demands of the MAV for fast response and lightweight in the tracking process. The algorithm procedure is shown in Figure 3.

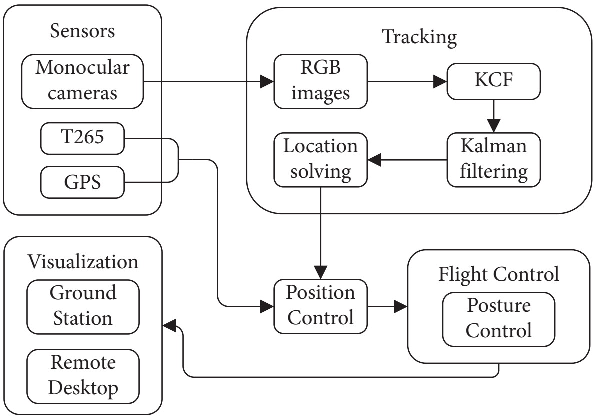


Figure 2: Software architecture of autonomous target tracking system.

To obtain more training samples, a training sample set is generated by the circulant matrix. The $n \times 1$ dimensional vector $x = [x_1, x_2, \ldots, x_n]^T$ is used as the basic sample, and the sample vector x is shifted by the permutation matrix L for n times. The training sample set of the current frame is formulated as follows:

$$X = T(x) = \begin{bmatrix} \left(L^0 x\right)^T \\ \left(L^1 x\right)^T \\ \left(L^2 x\right)^T \\ \vdots \\ \left(L^{n-1} x\right)^T \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix}. \tag{1}$$

The definition of the circulant matrix L is as follows:

$$L = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \tag{2}$$

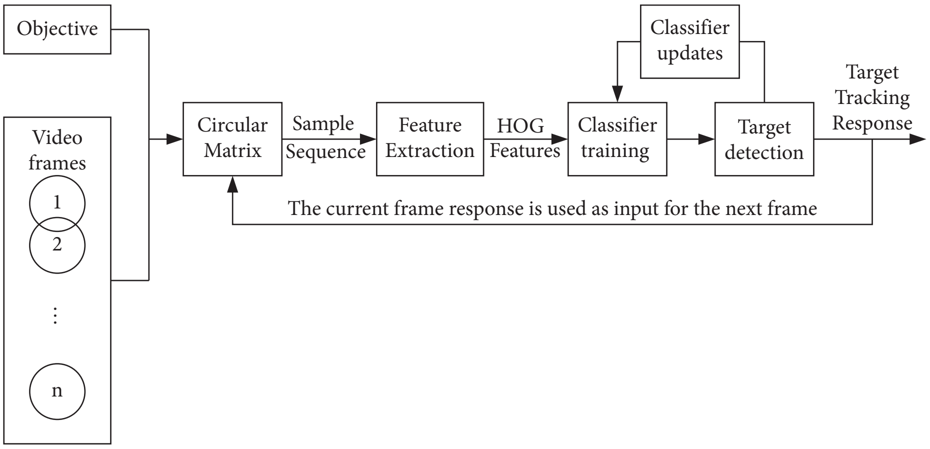


Figure 3: KCF target tracking algorithm flow chart.

In order to improve the calculation speed, the discrete Fourier matrix is used to diagonalize the sample set as follows:

$$X = F\mathrm{diag}\,(\widehat{x})F^H, \tag{3}$$

where $\widehat{x}$ is the discrete Fourier transform of the basic sample x, $\mathrm{diag}\,(\widehat{x})$ is the diagonal matrix, F is the Fourier matrix, and $F^H = (F^*)^T$ represents the complex conjugate transpose matrix.

We created the classifier $f\,(z) = \omega^T z$ with the ridge regression model, where z is the candidate sample. The goal is to minimize the squared error over training samples x_i and regression targets y_i, which can be written as follows:

$$\min_{\omega} \sum_i \left(f\,(x_i) - y_i\right)^2 + \lambda\|\omega\|^2, \tag{4}$$

where ω is the weight coefficient of the classifier and λ is the regularizing term coefficient. In order to improve the generalization ability of the classifier and prevent the overfitting phenomenon of the classifier, a regularizing term $\lambda\|\omega\|^2$ is used to control the overfitting.

By setting the partial derivative of ω to zero, the expression of ω is as follows:

$$\omega = \left(X^T X + \lambda E\right)^{-1} X^T y, \tag{5}$$

where E is the unit matrix and y is the column vector composed of the regression label y_i of each sample. We converted equation (5) into the complex field, which can be written as follows:

$$\omega = \left(X^H X + \lambda E\right)^{-1} X^H y. \tag{6}$$

Using the diagonalizable property of the circulant matrix, the expression of equation (6) in the frequency domain can be represented as follows:

$$\widehat{\omega} = \frac{\widehat{x} \odot \widehat{y}}{\widehat{x} \odot \widehat{x}^* + \lambda}, \tag{7}$$

where $\widehat{\omega}$ and $\widehat{y}$ represent the Fourier transform of ω and y, respectively, and $\widehat{x}^*$ represents the conjugate matrix of $\widehat{x}$.

As the target tracking is a nonlinear problem, the sample x can be mapped to a high-dimensional space through the mapping function $\varphi\ (x)$ to make the nonlinear problem linearly separable. The weight coefficient ω of the classifier can be expressed as follows:

$$\omega = \sum_i \alpha_i \varphi\,(x_i), \tag{8}$$

where α_i is the linear combination coefficient, and the kernel function k is defined as follows:

$$k\left(x, x^{'}\right) = k^{xx^{'}} = \varphi\,(x)^T \varphi\left(x^{'}\right). \tag{9}$$

The $n \times n$ dimensional kernel matrix K composed of kernel functions between the samples is expressed as follows:

$$K_{ij} = k\left(x_i, x_j\right). \tag{10}$$

Then, the ridge regression function can be expressed as follows:

$$f\,(z) = \sum_{i=1}^{n} \alpha_i k\,(z, x_i). \tag{11}$$

The expression of α can be derived as follows:

$$\alpha = (K + \lambda E)^{-1} y, \tag{12}$$

where α is a coefficient vector composed of α_i. The Fourier transform of equation (12) can be expressed as follows:

$$\widehat{\alpha} = \frac{\widehat{y}}{\widehat{k}^{xx} + \lambda}, \tag{13}$$

where $\widehat{\alpha}$ is the Fourier transform form of α and $\widehat{k}^{xx}$ is the Fourier transform form of the first row of matrix K.

After training the classifier with numerous samples obtained by the circulant matrix, the target can be detected and located. First of all, the kernel matrix K^z between the sample x and the candidate sample z is calculated to match the position results.

$$K^z = C\,(k^{zx}), \tag{14}$$

where $C(k^{zx})$ represents the circulant matrix of vector k^{zx}.

The regression function of the candidate sample is as follows:

$$f(z) = \left(K^{z}\right)^{T}\alpha. \tag{15}$$

Equation (15) is converted into the frequency domain, which can be expressed as follows:

$$\hat{f}(z) = \hat{k}^{xz} \odot \hat{\alpha}. \tag{16}$$

In particular, the Gaussian kernel $k(x, x') = \exp(-1/\sigma^2\|x - x'\|^2)$ is selected as the kernel function; the Gaussian kernel function can be obtained as follows:

$$k^{xx'} = \exp\left(-\frac{1}{\sigma^2}\left(\|x\|^2 + \left\|x'\right\|^2 - 2F^{-1}\left(\hat{x}^{*} \odot \hat{x}'\right)\right)\right). \tag{17}$$

By Fourier transforming, the matrix inversion process is avoided. The time complexity of the algorithm is reduced from O (n^2) to O ($n$log$n$), which realizes fast detection and reduces the dependence on computer performance.

3.2. Design of Target Tracking Algorithm Based on Kalman Filter. In the previous section, a good balance between speed and accuracy is achieved by using the KCF filter to track the target and obtain the target motion state while the camera is stationary. However, the UAV tracking target is a dynamic process and the position estimation based on the previous section is not robust enough for this process. During the tracking process, it is not guaranteed that the target is always within the field of view of the camera, and occasionally the target may be partially or fully occluded, leading to target loss. Although the complete loss of the target caused by long-term occlusion may not be solved, the proposed method can deal with small-scale occlusion problem in a short time. Based on the abovementioned situation, this section applies Kalman filter to establish the linear motion model of the target and fuses the tracking results of KCF, while considering camera jitter as Gaussian noise. According to the input and output of the model, the optimal estimation of the motion state of the target can predict the target motion position at the next moment, so as to improve the tracking accuracy and robustness.

The Kalman filter is widely applied in the state estimation of target motion [23–25]. Due to noise during the measurement of target motion, Kalman filter can effectively remove noise by using the motion information of the target and obtain the optimal estimation of the target position.

Firstly, due to the high sampling frequency of the camera, the time interval between adjacent frames of the image is very short, the motion of the target between two frames can be regarded as uniform motion, and the acceleration of the target obeys Gaussian distribution. The state space vector of the system can be expressed as follows:

$$x_k = \left[\, x_{ik} \;\; y_{ik} \;\; \dot{x}_{ik} \;\; \dot{y}_{ik} \,\right]^{T}, \tag{18}$$

$$u_k = \left[\, \ddot{x}_{ik} \;\; \ddot{y}_{ik} \,\right]^{T}, \tag{19}$$

where x_k and u_k are the state vector and control vector of the system at time k, respectively; x_{ik} and y_{ik} represent the position of the target at time k in I, respectively; $\dot{x}_{ik}$ and $\dot{y}_{ik}$ represent the velocity of the target at time k in I, respectively; and $\ddot{x}_{ik}$ and $\ddot{y}_{ik}$ represent the acceleration of the target at time k in I, respectively.

The motion state equation of the system is as follows:

$$x_k = A_k x_{k-1} + B_k u_k + w_k, \tag{20}$$

where A_k is the state transition matrix of the system at time k, $x_{k\text{-}1}$ is the state vector of the system at time k-1, B_k is the control input matrix of the system at time k, u_k is the control vector of the system at time k, and w_k is the noise of the system at time k.

Assuming that the motion of the UAV tracking target is uniform, the specific forms of A and B are as follows:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{21}$$

$$B = \begin{bmatrix} \dfrac{\Delta t^2}{2} & 0 & \Delta t & 0 \\ 0 & \dfrac{\Delta t^2}{2} & 0 & \Delta t \end{bmatrix}^{T}. \tag{22}$$

The KCF tracking result can be used as the observation of Kalman filter. The observation equation can be written as follows:

$$z_k = H_k x_k + v_k, \tag{23}$$

where z_k is the target tracking result at time k, H_k is the state observation matrix, and v_k is the measurement noise at time k.

The specific form of H is as follows:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{24}$$

During the process of estimation, Kalman filter can be divided into two stages: prediction stage and iterative update stage. The specific processes are as follows:

(1) Prediction stage

From the motion state equation,

$$\begin{aligned} \hat{x}_k^{-} &= A_k \hat{x}_{k-1} + B_k u_{k-1}, \\ P_k^{-} &= A_k P_{k-1} A_k^{T} + Q, \end{aligned} \tag{25}$$

where $\hat{x}_k^{-}$ is the prior state estimation of the target at time k, $\hat{x}_{k-1}$ is the posterior state estimation of the target at time k-1, P_k^{-} is the prior estimation covariance matrix, P_{k-1} is the optimal estimation covariance matrix, and Q is the process noise covariance matrix.

(2) Iterative update stage

$$\begin{aligned}K_k &= P_k^- H^T \left(HP_k^- H^T + R\right)^{-1},\\ \hat{x}_k &= \hat{x}_k^- + K_k \left(z_k - H\hat{x}_k^-\right),\\ P_k &= \left(E - K_k H\right)P_k^-,\end{aligned} \tag{26}$$

where K_k is the Kalman gain matrix, R is the measurement noise covariance matrix, and E is the unit matrix.

In summary, the tracking process based on the KCF and Kalman filter is shown in Figure 4. Firstly, the KCF target tracking algorithm and Kalman filter are initialized, and the target state prediction value at the current moment is calculated from the optimal estimation value of the target state at the previous moment. Then, the predicted covariance at the current time is calculated from the optimal estimated covariance matrix at the previous time and the process noise. In the update stage, the KCF algorithm is applied to track the selected target. After the target tracking result z_k is obtained, the forecasting result $\hat{x}_k^-$ is corrected by Kalman gain. Finally, the optimal estimate $\hat{x}_k$ of the current target state is obtained.

4. Three-Dimensional Position Solution

After obtaining the target's plane motion coordinates in the two-dimensional image from Section 3, the coordinates are converted into three-dimensional space using the following method, so that the UAV can track dynamically.

As shown in Figure 5, the world coordinate system, body coordinate system, camera coordinate system, image coordinate system, and pixel coordinate system are defined, and the relative motion relationship between the UAV and the target is described. Among them, $W = \{o_w, x_w, y_w, z_w\}$ is the world coordinate system, $B = \{o_b, x_b, y_b, z_b\}$ is the body coordinate system, $C = \{o_c, x_c, y_c, z_c\}$ is the camera coordinate system, $I = \{o_i, x_i, y_i\}$ is the image coordinate system, $G = \left\{o_g, u, v\right\}$ is the pixel coordinate system, and the unit is the pixel. The pixel coordinate system takes the left vertex of the image as the origin, u as right axis, and v as down axis.

Suppose the coordinate of the target point M in W is (x_w, y_w, z_w), the coordinate of its projection m in I is (x_i, y_i), and the coordinate of the origin o_i of I in G is (u_0, v_0). Then, the relationship between G and I can be expressed as follows:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \dfrac{1}{dx} & 0 & u_0 \\ 0 & \dfrac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}, \tag{27}$$

where dx and dy are the physical dimensions of the unit pixel on the x_i axis and the y_i axis, respectively.

Let the coordinate of the target point M in C be x_c, y_c, and z_c. According to the projection transformation, the relationship between I and C can be expressed as follows:

$$z_c \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}, \tag{28}$$

where f is the focal length of the camera, which is determined by the internal parameters of the camera.

Invoking equation (18) with equation (19), the relationship between G and C can be written as follows:

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}, \tag{29}$$

where $f_x = f/dx$ and $f_y = f/dy$ represent the horizontal pixel focal length and vertical pixel focal length, respectively, and let $S = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$ be the camera internal parameter matrix.

Then, the coordinate of M in W can be expressed as follows:

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = R_B^W R_C^B \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}, \tag{30}$$

where R_C^B is the transformation matrix from C to B, R_B^W is the rotation matrix from B to W, and r_{ij} is determined by the attitude angle of the UAV. The specific forms are as follows:

$$\begin{aligned} R_B^W &= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix},\\ R_C^B &= \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}. \end{aligned} \tag{31}$$

Invoking equation (21) with equation (20), the relationship between G and W can be written as follows

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = z_c R_B^W R_C^B S^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}. \tag{32}$$

After determining the coordinates of the target point M in the image sequence, its position coordinate in W can be calculated. However, as the monocular camera cannot obtain the depth information z_c, the similar triangle estimation method is used to estimate the depth information of the target. The premise of the estimation is to know the actual height of the target, so the height of the target is measured by the triangulation method. The triangulation method is shown in Figure 6.

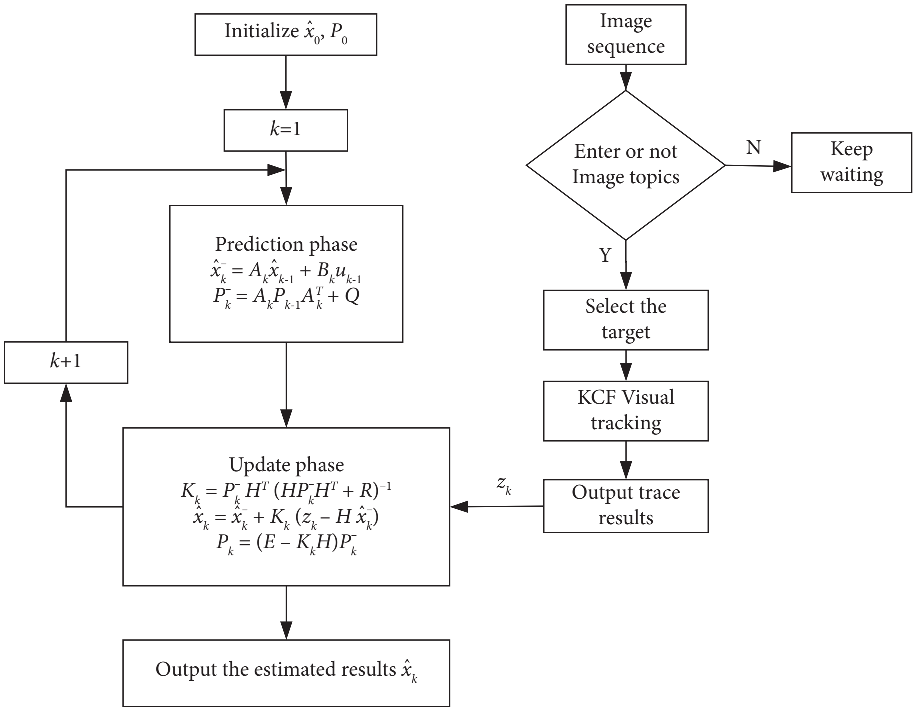

Initialize $\hat{x}_0$, P_0
k=1
Prediction phase
$\hat{x}_k^- = A_k\hat{x}_{k-1} + B_k u_{k-1}$
$P_k^- = A_k P_{k-1} A_k^T + Q$
k+1
Update phase
$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}$
$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H\hat{x}_k^-)$
$P_k = (E - K_k H)P_k^-$
Output the estimated results $\hat{x}_k$
Image sequence
Enter or not Image topics
N
Keep waiting
Y
Select the target
KCF Visual tracking
Output trace results
z_k

Figure 4: Tracking algorithm flow chart.

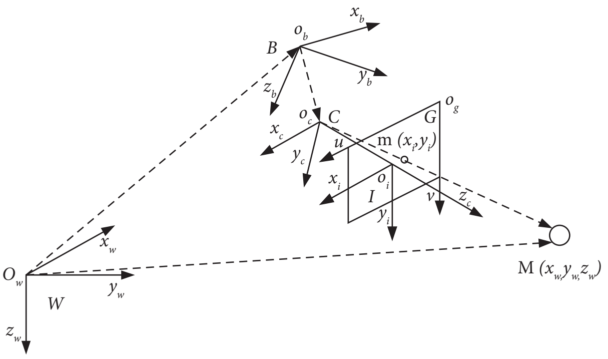


Figure 5: Position relationship between the UAV and the target.

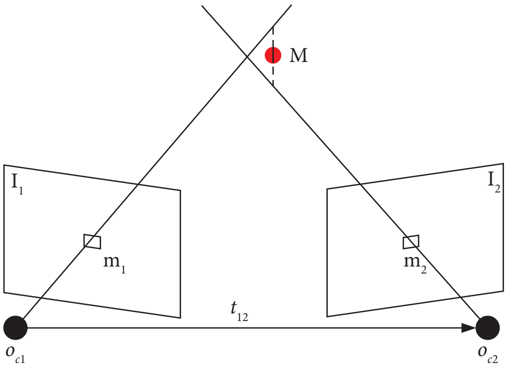


Figure 6: Triangulation method.

For images I_1 and I_2, with the left image as a reference, the camera optical centre moves horizontally from o_{c1} to o_{c2}. During the movement, it is assumed that the camera does not rotate and the displacement of the z_c axis and y_c axis are negligible. Suppose I_1 has the feature point m_1 and its coordinate in C is x_{c1}, y_{c1}, and z_{c1}. The feature point in I_2 is m_2, and its coordinate in C is x_{c2}, y_{c2}, and z_{c2}. According to the definition of epipolar geometry [26], the coordinate relationship can be expressed as follows:

$$z_{c2}P_{c2} = z_{c1}P_{c1} + t_{12}, \tag{33}$$

where $P_{c1} = [\, x_{c1}/z_{c1} \;\; y_{c1}/z_{c1} \;\; 1\,]^T$ and $P_{c2} = [\, x_{c2}/z_{c2} \;\; y_{c2}/z_{c2} \;\; 1]^T$ are respectively the normalized coordinates of m_1 and m_2 in C, t_{12} is the translation vector from o_{c1} to o_{c2}, and its value is known. Left multiply $P_{c2}^{\wedge}$ on both sides of the equation, where ˆ represents the outer product operation, and the following relationship is formulated as follows:

$$z_{c2}P_{c2}^{\wedge}P_{c2} = 0 = z_{c1}P_{c2}^{\wedge}P_{c1} + P_{c2}^{\wedge}t_{12}. \tag{34}$$

According to the right side of the equation, z_{c1} can be calculated, and the depth value of the target in I_1 can be calculated. The actual height of the target is calculated according to the similar triangle, as shown in Figure 7.

Assuming that H_m is the actual height of the target, h_m is the height of the target in the image, then H_m can be expressed as follows:

$$H_m = \frac{z_{c1} \times h_m}{f}. \tag{35}$$

After estimating H_m based on the first two frames, the depth value z_c of the target in the subsequent frames is formulated by the similarity relationship as follows:

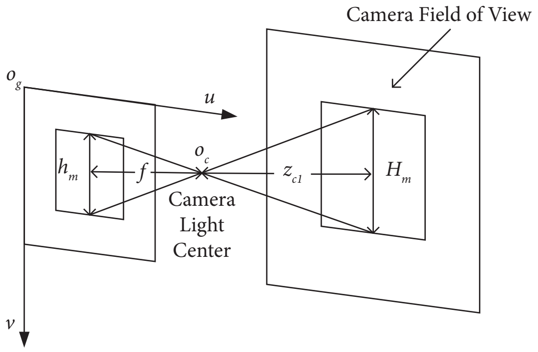


Figure 7: Similarity estimation.

$$z_c = \frac{H_\mathrm{m} \times f}{h_\mathrm{m}}. \tag{36}$$

5. Experiment and Analysis

The flight experiment was carried out in an open outdoor environment. During the experiment, As the UAV and the target are in motion, the difficulty of pose estimation is increased. In addition, during the occlusion experiment, the flight parameters of the UAV are set to prevent the UAV from large-scale manoeuvring in this paper. The flight parameters are shown in Table 1.

First, the ground station is applied to check the sensor data of the UAV after power on. Then, the UAV is switched to fixed-point mode by using a 2.4 GHz remote controller, and the UAV is unlocked and controlled to hover at a fixed point after taking off to a certain height. After selecting the tracking target, in order to estimate the three-dimensional position coordinate of the target, the size of the target is first measured and estimated to provide a reference for subsequent depth estimation. In this paper, the sizes of three different types of targets are estimated. The matching results are shown in Figure 8, and the estimation results are shown in Table 2.

It can be seen from Table 2 that the proposed estimation method can effectively estimate the size of different types of targets. The estimation errors are within 100 mm, which is completely acceptable for depth estimation. To verify the depth estimation algorithm proposed in this paper, targets with different distances are selected for depth estimation.

Table 3 shows the estimated distances of Person, Car, and UAV at different distances. It can be seen that the estimation errors of the algorithm are within 0.2, and the estimation errors do not change greatly with the increase of distance. After that, the target tracking experiment can be carried out.

As the tracking process is processed in real time on an onboard computer, the outputs of the tracking system send control instructions to the flight control system through serial communication. Limited by the processing speed of the onboard computer, this paper uses the remote control to make the flight control system enter the Offboard mode when switching the Offboard mode. The tracking algorithm is automatically started to track the target when the target is selected. The first perspective tracking view of the UAV is shown in Figure 9, where the green border is the KCF tracking result, and the yellow border is the Kalman forecasting result.

Table 1: List of flight parameters.

Parameter	Value
x-axis maximum flight velocity	0.5 m/s
y-axis maximum flight velocity	0.5 m/s
z-axis maximum rising velocity	1.0 m/s
Maximum landing velocity	0.5 m/s
Maximum yaw angular velocity	15 deg/s
Safety fence radius	10 m

Tracking experimental results of target occlusion are shown in Figure 10. It can be seen from the results that even when the tracked target is completely occluded or partially occluded, the KCF tracking result will drift, but the algorithm proposed in this paper can still track the target effectively.

When the tracking target is occluded, using only the KCF algorithm results in significant position estimation errors. However, using the KCF algorithm to fuse the Kalman filter, the errors are within the allowable range. The experimental results are shown in Figure 11.

In Figure 11, the tracking target is occluded at 120 s and 220 s. It can be clearly seen that the proposed algorithm improves the tracking effect in the occlusion process and effectively reduces the position estimation errors of the target. The position estimation error of the x-axis and y-axis is reduced from about 0.8 m to about 0.3 m, and the position estimation error of z-axis is reduced from about 0.2 m to 0.1 m.

To further evaluate the system, the dynamic position of the target and the estimated results are compared, as shown in Figure 12. The system can effectively estimate the position of the target in three-dimensional space for most of the time. Despite jitter and occasional drift, the proposed algorithm can still relocate the target in a short time.

The errors between target position and estimated position in x-, y-, and z-axes are shown in Figure 13. For most of the time, the errors of the estimation results on the x-axis are mostly kept within 0.6 m, and the errors on the y-axis and z-axis are kept within 0.2 m. The RMSE (root mean square error) and MAE (mean absolute error) are further calculated, and the results are shown in Table 4. The experimental results show that the proposed algorithm can track the target effectively.

Compared with the 3D target pose estimation system in the paper [27], it is robust enough for real-time dynamic position estimation. In addition, in order to analyze the effect of the distance between the UAV and the target object on the accuracy of the target position estimation, several of target trajectory estimation experiments were performed. As shown in Table 5, it can be concluded that the performance of the proposed method does not deteriorate significantly when the distance between the UAV and the tracking object increases.

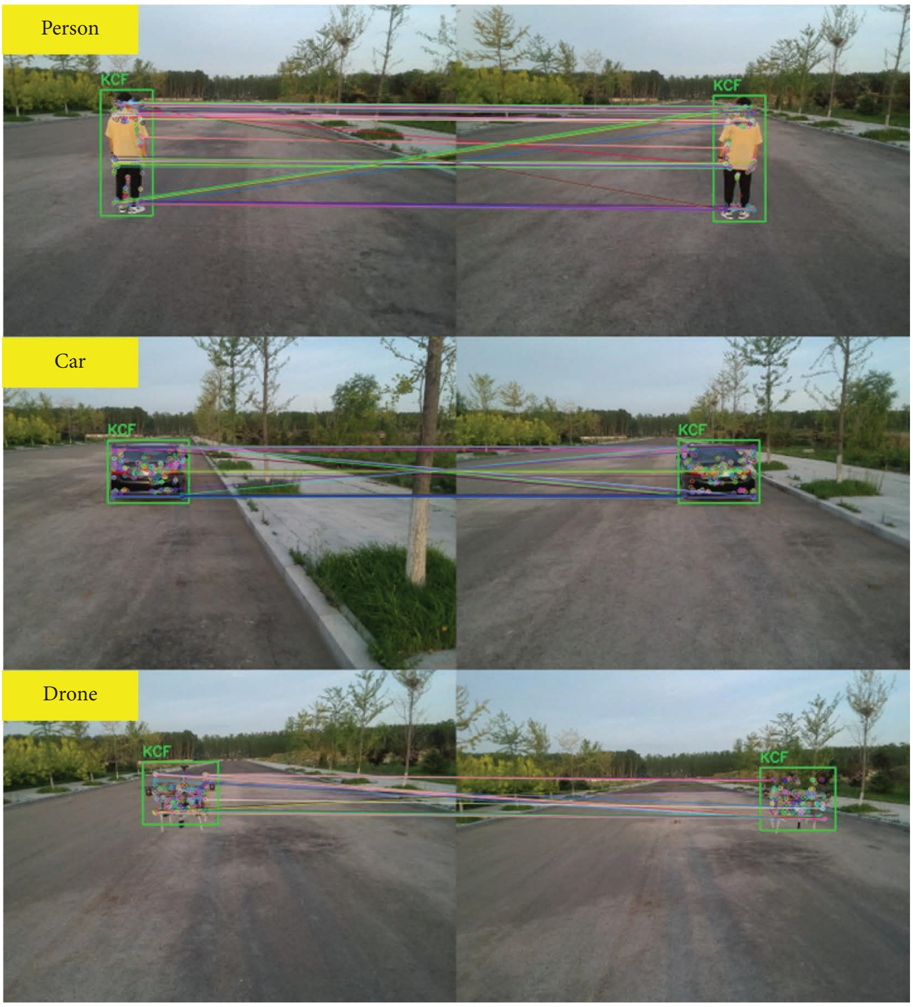


Figure 8: Matching results.

Table 2: Target size estimation results.

Target type	Actual size (mm)	Estimated size (mm)	Error (mm)
Person	1750 × 500	1816 × 587	66 × 87
Car	1450 × 1859	1523 × 1957	73 × 98
Drone	400 × 450	484 × 527	84 × 77

Table 3: Target depth estimation results.

Target type	Actual distance (m)	Estimated distance (m)	Actual distance (m)	Estimated distance (m)	Actual distance (m)	Estimated distance (m)
Person	2.00	2.08	3.00	3.12	4.00	4.15
Car	2.00	2.11	3.00	3.11	4.00	4.18
Drone	2.00	2.17	3.00	3.18	4.00	4.20

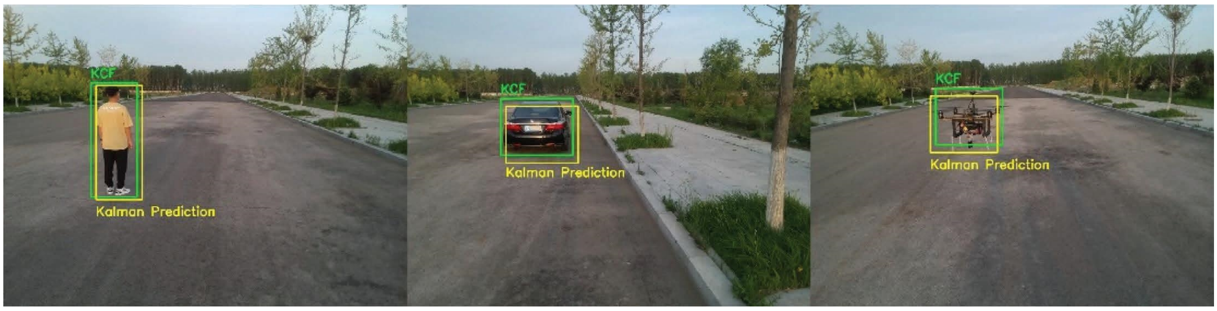


Figure 9: UAV first perspective tracking view.

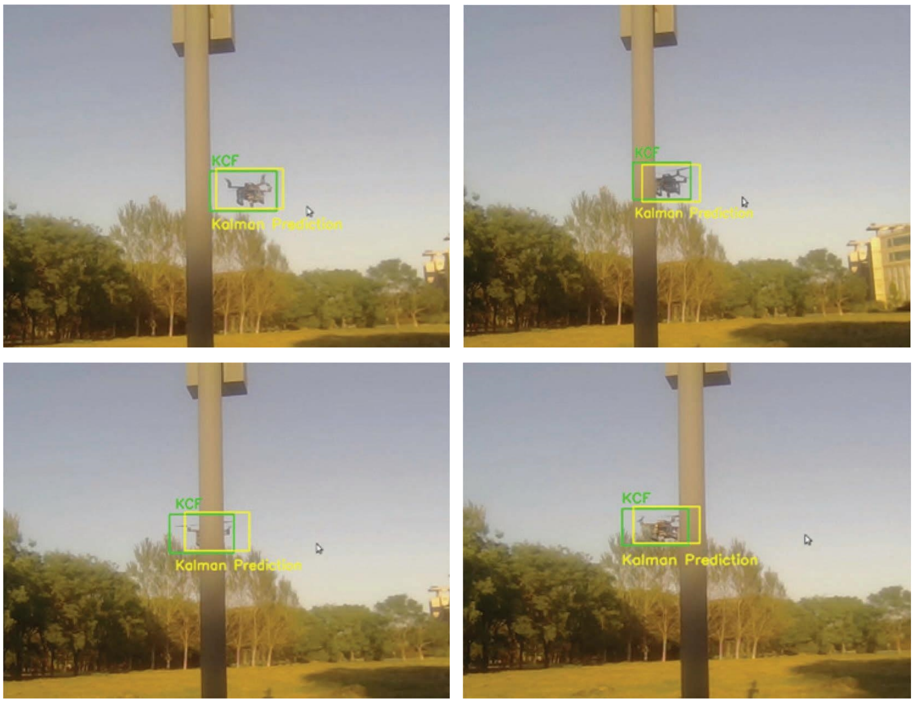
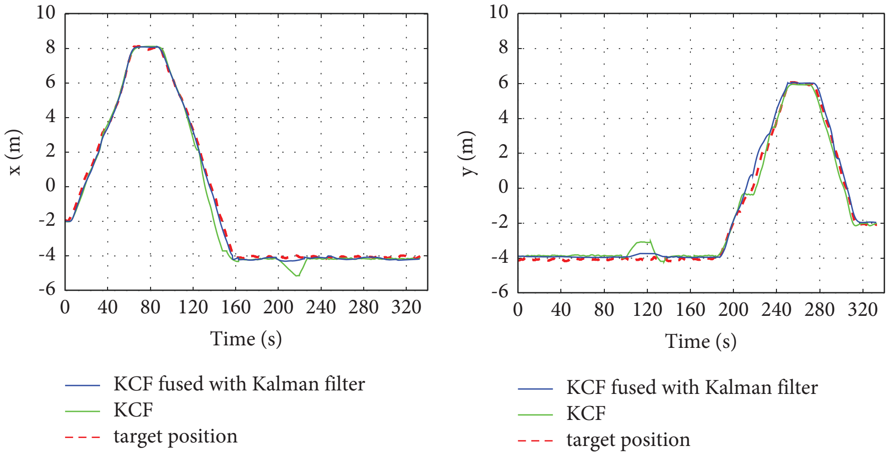


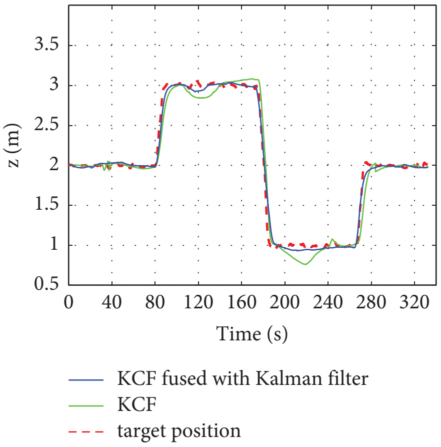
Figure 10: Tracking experimental results of target occlusion.



(a)

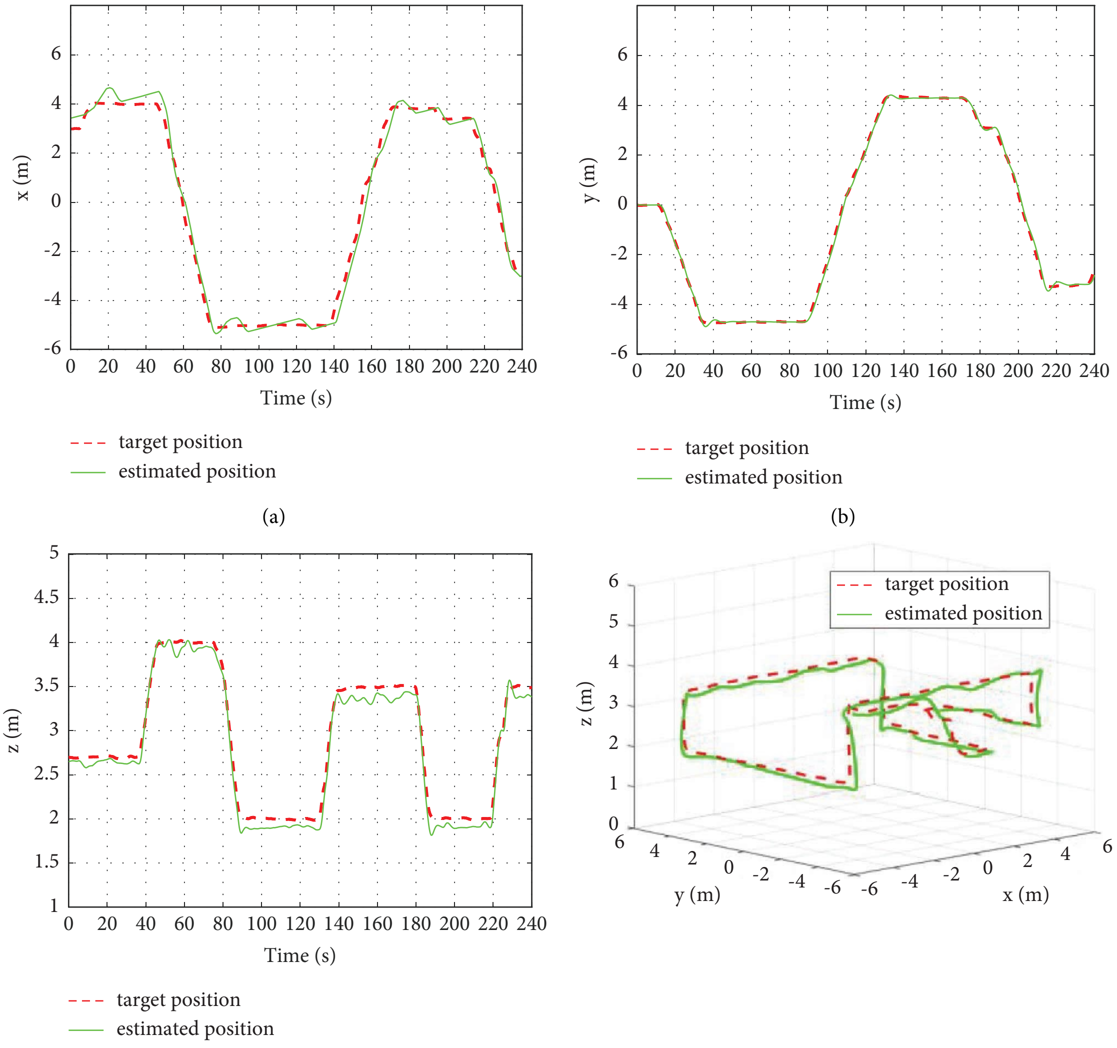
(b)

Figure 11: Continued.



(c)

Figure 11: Target position and estimated position results: (a) x-axis position. (b)y-axis position. (c) z-axis position.



(a)

(b)

(c)

(d)

Figure 12: Target position and estimated position results: (a) x-axis position. (b)y-axis position. (c) z-axis position. (d) Three-dimensional trajectory.

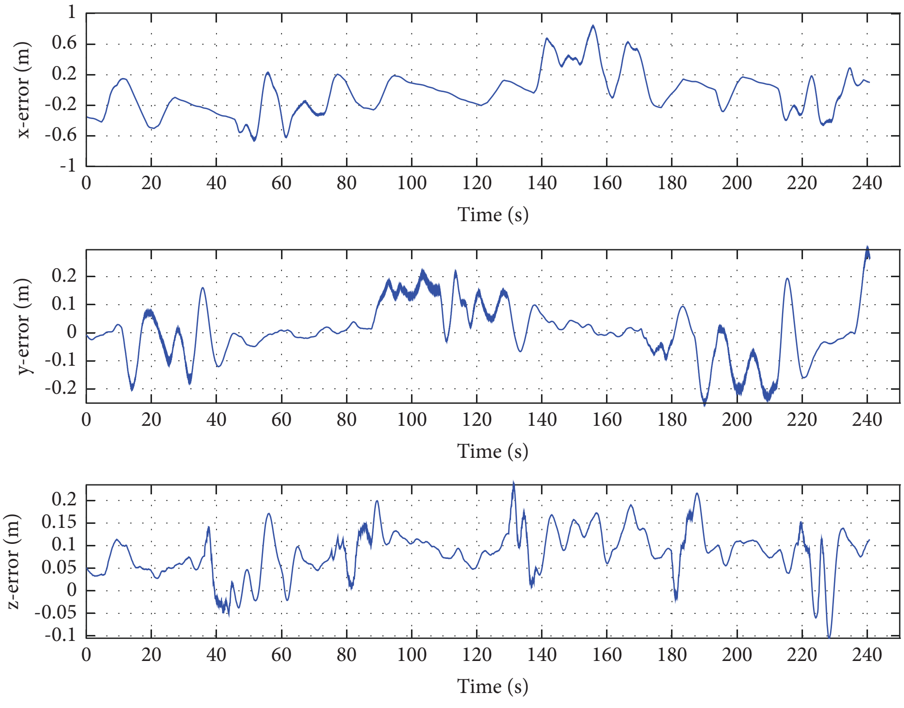


Figure 13: Error between target position and estimated position in x-, y-, and z-axes.

Table 4: RMSE and MAE of dynamic target position estimation.

Estimated error	x (m)	y (m)	z (m)
RMSE	0.2764	0.0947	0.0985
MAE	0.2181	0.0689	0.0884

Table 5: RMSE and MAE of dynamic target position estimation at different distances.

Object distance	1–3 m			8–10 m		
Estimated error	x (m)	y (m)	z (m)	x (m)	y (m)	z (m)
RMSE	0.2764	0.0947	0.0985	0.3142	0.1324	0.1238
MAE	0.2181	0.0689	0.0884	0.2647	0.1205	0.1056

6. Conclusion

The payload and endurance of MAV are limited, and it is impossible to carry a large onboard computer to run complex visual tracking algorithms. Aiming at the above problems, this paper proposes a MAV target tracking algorithm based on monocular vision. The main contributions are as follows:

(1) For the problem of measuring the distance between the MAV and the target, a triangulation algorithm has been designed for a monocular camera to estimate the object's size. Based on this, the triangle similarity can measure the distance between the micro-MAV and target;

(2) To address the problem of target occlusion, the paper proposes a target tracking algorithm based on KCF and Kalman filter. The algorithm combines the tracking results with the Kalman filter, solving the short-term occlusion problem and improving the anti-interference ability in the tracking process;

(3) The proposed target tracking algorithm is evaluated through numerous experiments in a real environment. The experimental results demonstrate the feasibility and robustness of the proposed algorithm.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the Liaoning Provincial Education Department Project (Grants no. LJKMZ20220614).

References

[1] J. Kim, S. Kim, C. Ju, and H. Son, “Unmanned aerial vehicles in agriculture: Unmanned Aerial Vehicles in Agriculture: A Review of Perspective of Platform, Control, and Applications review of perspective of platform, control, and applications,” *IEEE Access*, vol. 7, pp. 105100–105115, 2019.

[2] Z. Zuo, C. Liu, Q. L. Han, and J. Song, “Unmanned aerial vehicles: control methods and future challenges,” *IEEE/CAA Journal of Automatca Sinica*, vol. 99, pp. 1–14, 2022.

[3] P. K. Reddy Maddikunta, S. Hakak, M. Alazab et al., “Unmanned aerial vehicles in smart agriculture: Unmanned Aerial Vehicles in Smart Agriculture: Applications,

Requirements, and Challengesppplications, requirements, and challenges," *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17608–17619, 2021.

[4] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges review, solutions, and challenges," *Computer Communications*, vol. 149, pp. 270–299, 2020.

[5] X. Wu, W. Li, D. T. Hong, R. Tao, Q. Du, and Q. Du, "Deep learning for unmanned aerial vehicle-based object detection and tracking: Deep Learning for Unmanned Aerial Vehicle-Based Object Detection and Tracking: A survey survey," *IEEE Geoscience and Remote Sensing Magazine*, vol. 10, no. 1, pp. 91–124, 2022.

[6] M. Ilhan, M. Tayyip Gürbüz, S. Acarer, and S. Acarer, "Unified low-pressure compressor concept for engines of future high-speed micro-unmanned aerial vehicles," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 233, no. 14, pp. 5264–5281, 2019.

[7] D. A. Mercado-Ravell, P. Castillo, and R. Lozano, "Visual detection and tracking with UAVs, following a mobile object," *Advanced Robotics*, vol. 33, no. 7-8, pp. 388–402, 2019.

[8] X. Liu, Y. Yang, C. Ma, J. Li, and S. Zhang, "Real-time visual tracking of moving targets using a low-cost unmanned aerial vehicle with a 3-axis stabilized gimbal system," *Applied Sciences*, vol. 10, no. 15, pp. 5064–5091, 2020.

[9] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Liu, "Visual object tracking using adaptive correlation filters," in *Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1500–1514, San Francisco, USA, June 2010.

[10] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-High-Speed Tracking with Kernelized Correlation Filterspeed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[11] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proceedings of the British Machine Vision Conference*, pp. 1–11, Nottingham, UK, September 2014.

[12] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proceedings of 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1–10, Venice, Italy, October 2017.

[13] K. N. Dai, D. Wang, H. C. Lu, C. Sun, and J. H. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4670–4679, Long Beach, CA, June 2019.

[14] C. Ma, J. B. Huang, X. Yang, and M. H. Yang, "Robust Robust Visual Tracking via Hierarchical Convolutional Featuresisual tracking via hierarchical convolutional features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 11, pp. 2709–2723, 2019.

[15] Z. Y. Huang, C. H. Fu, Y. M. Li, and F. L. Lin, "Learning aberrance repressed correlation filters for real-time UAV tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2891–2900, Seoul, South Korea, October 2019.

[16] Z. L. Qiu, Y. Zha, P. Zhu, and M. Wu, "Visual tracking algorithm based on online feature discrimination with Siamese network," *Acta Optica Sinica*, vol. 39, pp. 253–261, 2019.

[17] A. Dhillon and G. K. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence*, vol. 9, no. 2, pp. 85–112, 2020.

[18] J. Zhang, J. Sun, J. Wang, and X. G. Yue, "Visual object tracking based on residual network and cascaded correlation filters," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 8, pp. 8427–8440, 2021.

[19] N. An and W. Qi Yan, "Multitarget tracking using Siamese neural networks," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 17, no. 2s, pp. 1–16, 2021.

[20] L. Bertinetto, J. Valmadre, J. F. Henriques, and A. Vedaldi, "Fully-convolutional siamese networks for object tracking," in *Proceedings of the 14th European Conference on Computer Vision (ECCV)*, pp. 1–15, Amsterdam, Netherlands, October 2016.

[21] H. Fan and H. Ling, "Siamese cascaded region proposal networks for real-time visual tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7952–7961, Long Beach, CA, USA, June 2019.

[22] M. X. Yu, Y. H. Zhang, Y. K. Li, J. Z. Li, and C. L. Wang, "Distractor-aware long-term correlation tracking based on information entropy weighted feature," *IEEE Access*, vol. 8, pp. 29417–29429, 2020.

[23] Y. Guo, D. Yang, and Z. Chen, "Object tracking on satellite videos: Object Tracking on Satellite Videos: A Correlation Filter-Based Tracking Method With Trajectory Correction by Kalman Filter correlation filter-based tracking method with trajectory correction by Kalman filter," *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 9, pp. 3538–3551, 2019.

[24] A. Arjas, E. J. Alles, E. Maneas et al., "Neural network kalman filtering for 3-d object tracking from linear array ultrasound data," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 69, no. 5, pp. 1691–1702, 2022.

[25] I. A. Iswanto, T. W. Choa, and B. Li, "Object tracking based on meanshift and particle-kalman filter algorithm with multi features," *Procedia Computer Science*, vol. 157, pp. 521–529, 2019.

[26] G. J. Wang, C. C. Feng, X. W. Hu, H. Wang, and H. Z. Yang, "Epipolar Epipolar Geometry Guided Highly Robust Structured Light 3D Imagingeometry guided highly robust structured light 3D imaging," *IEEE Signal Processing Letters*, vol. 28, pp. 887–891, 2021.

[27] Y. R. Feng, K. Tse, S. Y. Chen, C. Y. Wen, and B. Li, "Learning-Based Autonomous UAV System for Electrical and Mechanical (E&amp; M) Device Inspectionm device inspection," *Sensors*, vol. 21, no. 4, pp. 1385–1404, 2021.